

# L'ESSENTIEL SUR MERISE 2

## 1 GENERALITES : 2

- 1.1 QU'EST-CE QUE MERISE ? 2
- 1.2 HISTORIQUE : 2
- 1.3 PRESENTATION : 2
- 1.4 SCHEMA DIRECTEUR : 3
- 1.5 LES SYSTEMES D'INFORMATION : 4
  - 1.5.1 Architecture & conception d'un système d'information 4
  - 1.5.2 Système d'information et système informatique 5
- 1.6 LE MODELE CONCEPTUEL DE LA COMMUNICATION : 7
  - 1.6.1 Définition de l'organisation 7
  - 1.6.2 Diagramme de contexte 7
  - 1.6.3 Diagramme conceptuel de flux 7

## 2 LA DEMARCHE DE LA METHODE MERISE : 8

- 2.1 GENERALITES : 8
  - 2.1.1 3 axes : 8
  - 2.1.2 Les trois niveaux d'analyse 8
  - 2.1.3 La méthode MERISE 9
- 2.2 L'ANALYSE (OU FORMALISATION) CONCEPTUELLE : 9
  - 2.2.1 Introduction : 9
  - 2.2.2 MCD (Modèle Conceptuel de Données) 9
    - 2.2.2.1 Description statique du système d'information 9
    - 2.2.2.2 Les concepts de base 10
  - La propriété (ou attribut ou rubrique) 10
  - 2.2.3 1.1) La propriété (ou attribut ou rubrique) 10
  - 2.2.4 1.2) L'entité ou individu-type 11
    - 2.2.4.1 a) Définition 11
    - 2.2.4.2 b) Les identifiants 12
    - 2.2.4.3 c) Occurrence d'entité ou individu 12
    - d) Notion de dépendance fonctionnelle directe 13
    - 2.2.4.4 Exercice n°1 (Acquis : Propriété et Entité) 13
  - 2.2.5 1.3) L'association (ou relation-type) 17
    - 2.2.5.1 a) Définition 17
    - 2.2.5.2 b) Occurrence d'association 18
    - 2.2.5.3 c) Cardinalité 19
    - 2.2.5.4 Exercice n°2 (Acquis : propriété, entité et association) 20
    - 2.2.5.5 d) Caractéristiques d'une association 23
    - 2.2.5.6 e) L'association porteuse 25
    - 2.2.5.7 Auto-évaluation n° 1: 25
    - 2.2.5.8 f) Les associations transitives 28
    - 2.2.5.9 g) Les associations réflexives 29
    - 2.2.5.10 Exercice n°3 (Acquis : Concepts de base uniquement) 30
    - 2.2.5.11 Auto-évaluation n° 2: 31
  - 2.2.6 MCT (Modèle dynamique : prg 2ème année) Modèle Conceptuel des Traitements 34
- 2.3 L'ANALYSE (OU FORMALISATION) ORGANISATIONNELLE : 34
  - 2.3.1 MOT Modèle Organisationnel des Traitements (2ème année) 34
  - 2.3.2 MLD Modèle Logique de Données. 35
- 2.4 L'ANALYSE (OU FORMALISATION) OPERATIONNELLE : 35
  - 2.4.1 MOPT (2ème année) Modèle organisationnel des traitements. 35
  - 2.4.2 MPD Modèle Physique des Données. 35

# L'ESSENTIEL SUR MERISE

## 1 GENERALITES:

### 1.1 *Qu'est-ce que Merise ?*

C'est une méthode d'analyse élaborée pour l'informatisation des systèmes d'information.

### 1.2 *Historique :*

Elle fut créée en 1978 sous l'impulsion du **Ministère de la Recherche et de l'industrie**. Dès 1980, c'est devenu un standard dans le domaine des systèmes intégrés de gestion. Ce système existe seulement dans les pays francophones (au Canada, il existe cependant des variantes).

### 1.3 *Présentation :*

Le système à une double vocation :

- Représenter une méthode de conception des systèmes d'information
- Proposer une démarche méthodologique du développement du système d'information.

Cette méthode est une analyse objective et critique de l'assistant à travers les enquêtes pour recueillir les éléments nécessaires à une **modélisation conceptuelle de données (MCD)** et des traitements de l'organisation.

Le responsable du projet mène cette étude avec la collaboration des individus et des groupes pour connaître leurs besoins.

A partir de ces éléments et des règles d'organisation, les responsables proposent une ou plusieurs solutions générales basées sur une nouvelle règle de gestion ne remettant pas en cause celle de départ.

La proposition de cette solution doit être accompagnée:

- \* de solides justificatifs pour convaincre le décideur
- \* d'une estimation des coûts et délais des solutions envisagées.

Notons :

- Cette méthode est applicable aux projets de toutes tailles
- Cette méthode est applicable aux secteurs public et privé.

## 1.4 Le Schéma directeur :

Cette **étape est inévitable** pour pouvoir concevoir le système MERISE

Il faut :

- \* Une étude préalable
- \* Un scénario de développement d'une organisation, une administration
- \* Une période de travail (recherche de solutions, développement, ...)
- \* Une double finalité: une comptabilité entre l'approche globale et la réaction étalée dans le temps

SANS OUBLIER la cohérence des différents projets dans le temps

- \* Trois facteurs : une **stratégie**, une **technologie**, une **organisation et un système d'information**

Pour faire un schéma directeur, il faut une analyse de l'environnement; grâce à celle-ci, on peut mettre les contraintes en évidence, les menaces et les opportunités ; et ensuite faire apparaître les ressources et les compétences de l'organisation.

On établit ainsi :

- \* Les différents facteurs clefs de succès
- \* Les objectifs de ces différents facteurs clefs de succès
- \* Les tâches et les activités
- \* Les informations relatives à ces tâches.

On identifie les processus autour desquels se situent les informations. Ensuite, on identifie et hiérarchise les systèmes d'information correspondants. On obtient ainsi, un schéma directeur.

**Il faut donc 5 phases pour le schéma directeur de MERISE.**

### 1. Lancement et initialisation.

On recherche un chef de projet et on constitue le groupe de projet qui doit veiller au bon déroulement de l'organisation, à la bonne concertation et coordination. On fixe les objectifs et on planifie les actions. On met en route le **Comité de Pilotage** qui détermine les travaux, les orientations et les solutions, et le **Groupe des utilisateurs** dont le rôle est dans la conception du schéma directeur (échantillonnage de personnes à interviewer).

#### 2. Analyse des entretiens et des enquêtes.

On détermine un diagnostic de l'organisation et des systèmes d'information. Puis, on précise les objectifs et les orientations et enfin, on établit le dessin de plusieurs scénarios.

#### 3. Conception des cibles et des trajectoires.

Cette **phase est la plus créative**.

En effet, on formalise le modèle conceptuel de traitement ( MCT ) sur chacun des scénarios. On recherche des solutions techniques et organisationnelles et on étudie les trajectoires à tenir.

A la fin de la phase, on doit obtenir une seule trajectoire.

## 4. Etablissement d'un plan d'action.

Dans le scénario retenu, on établit le plan d'action avec tous les projets retenus et leurs établissements correspondants.

On évalue de manière globale l'ensemble des projets.

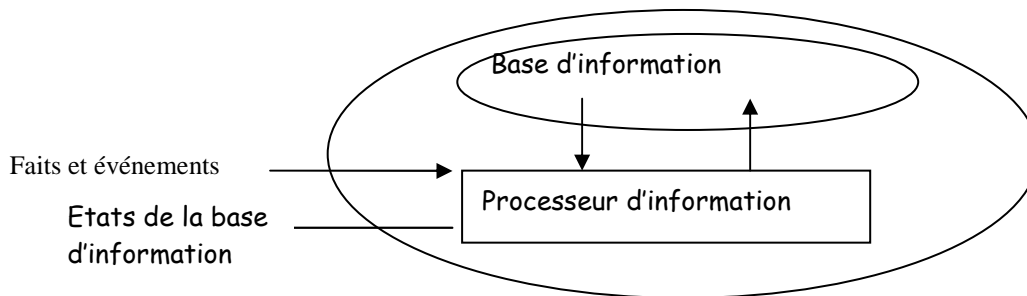
## 5. Procédure de suivi et de contrôle.

On met en place des structures de suivi et d'actualisation. On projette les réunions qui auront lieu périodiquement.

### 1.5 Les systèmes d'information :

#### 1.5.1 Architecture & conception d'un système d'information

Le système d'information doit décrire (on dit encore représenter) le plus fidèlement possible le fonctionnement du système opérant. Pour ce faire, il doit intégrer une base d'information dans laquelle seront mémorisés la description des objets, des règles et des contraintes du système opérant. Cette base étant sujette à des évolutions, le système d'information doit être doté d'un mécanisme (appelé processeur d'information) destiné à piloter et à contrôler ces changements. Le schéma suivant synthétise l'architecture d'un système d'information.



Le processeur d'information produit des changements dans la base d'information à la réception d'un message. Un message contient des informations et exprime une commande décrivant l'action à entreprendre dans la base d'information. Le processeur d'information interprète la commande et effectue le changement en respectant les contraintes et les règles.

Si le message exprime une recherche sur le contenu de la base d'information, le processeur interprète la commande et émet un message rendant compte du contenu actuel de la base d'information. Dans tous les cas, l'environnement a besoin de connaître si la commande a été acceptée ou refusée. Le processeur émet, à cet effet, un message vers l'environnement.

Relativement à la conception d'un système d'information, l'architecture présentée ci-dessus induit une double conception :

celle de la base d'information (aspect statique)

celle du processeur de traitement (aspect dynamique)

Pour aider le concepteur dans ces deux tâches, la méthode Merise propose un ensemble de formalismes et de règles destinées à modéliser de manière indépendante les données et les traitements du système d'information. Ces modèles ne sont qu'une base de réflexion pour le concepteur et un moyen de communication entre les divers acteurs du système d'information dans l'entreprise. Seul la validation de l'ensemble se fera en commun.

## 1.5.2 Système d'information et système informatique

Parmi les informations qui appartiennent au système d'information, certaines doivent ou peuvent faire l'objet d'un traitement automatisé grâce aux outils informatiques. Pour assurer la cohérence du système d'information, la méthode Merise propose une démarche d'informatisation comportant les étapes suivantes :

**le schéma directeur** : dont le rôle est de définir, de manière globale, la politique d'organisation et d'automatisation du système d'information. Pour ce faire, il est nécessaire de répertorier l'ensemble des applications informatiques existantes à modifier et à développer. Pour rendre contrôlable et modulable ce développement, il est nécessaire de découper le système d'information en sous-ensembles homogènes et relativement indépendant. Ces sous-ensembles sont appelés domaines. *Par exemple, on peut trouver le domaine « Approvisionnement », le domaine « Personnel ».* Les résultats attendus à la fin de cette étape sont une définition précise des domaines, une planification du développement de chaque domaine et un plan détaillé, année par année, des applications qui doivent être réalisées.

**l'étude préalable par domaine**: qui doit aboutir à une présentation générale du futur système de gestion (modèles des données et des traitements) en indiquant les principales novations par rapport au système actuel, les moyens matériels à mettre en œuvre, les bilans coût - avantage. Cette étude est réalisée en 4 phases :

une **phase de recueil** qui a pour objectif d'analyser l'existant afin de cerner les dysfonctionnements et les obsolescences les plus frappantes du système actuel.

une **phase de conception** qui a pour objectif de formaliser et hiérarchiser les orientations nouvelles en fonction des critiques formulées sur le système actuel et d'autre part des politiques et des objectifs de la direction générale. Cela revient à modéliser le futur système avec une vue pertinente de l'ensemble.

une **phase d'organisation** dont l'objectif est de définir le système futur au niveau organisationnel: qui fait quoi ?

une **phase d'appréciation** dont le rôle est d'établir les coûts et les délais des solutions définies ainsi que d'organiser la mise en œuvre de la réalisation. A cet effet un découpage en projets est effectué.

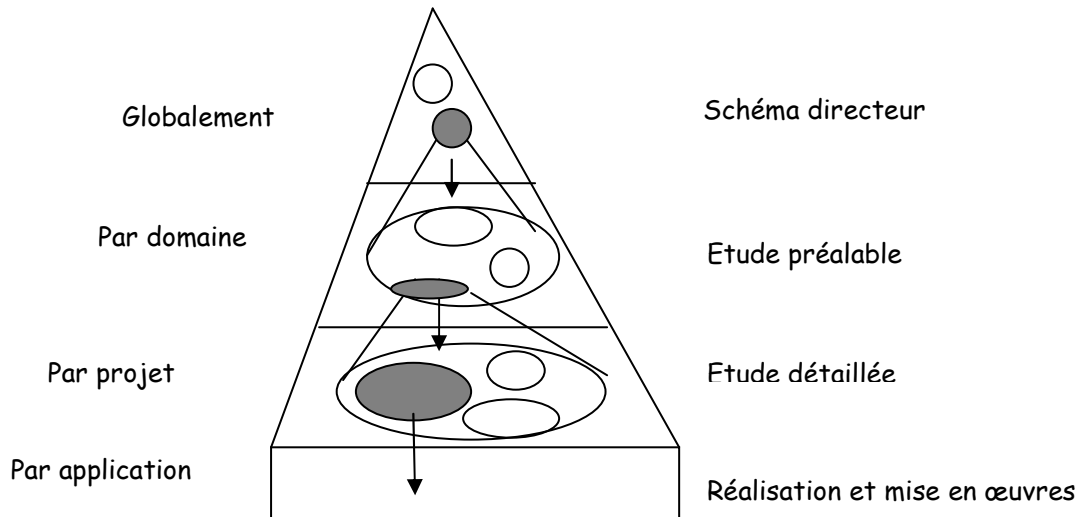
**l'étude détaillée par projet** qui consiste d'une part à affiner les solutions conçues lors de l'étude préalable et d'autre part à rédiger, pour chaque procédure à mettre en œuvre, un dossier de spécifications détaillé décrivant les supports (maquettes d'états ou d'écran) ainsi que les algorithmes associés aux règles de gestion... A l'issue de cette étude, il est possible de définir le cahier des charges utilisateurs qui constitue la base de l'engagement que prend le concepteur vis à vis des utilisateurs. Le fonctionnement détaillé du futur système, du point de vue de l'utilisateur, y est entièrement spécifié.

**la réalisation** dont l'objectif est l'obtention des programmes fonctionnant sur un jeu d'essais approuvés par les utilisateurs.

**la mise en œuvre** qui se traduit par un changement de responsabilité : l'équipe de réalisation va en effet transférer la responsabilité du produit à l'utilisateur. Cette étape intègre en particulier la formation des utilisateurs. Après une période d'exploitation de quelques mois, la recette définitive de l'application est prononcée.

**la maintenance** qui consiste à faire évoluer les applications en fonction des besoins des utilisateurs, de l'environnement et des progrès technologiques.

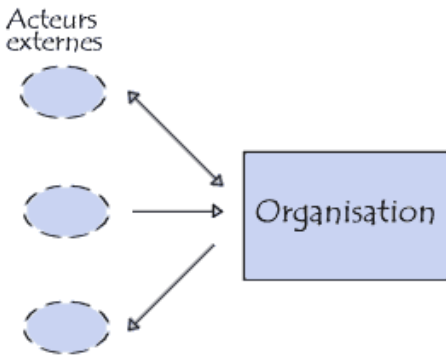
Le schéma suivant, extrait de l'ouvrage « *La méthode Merise* » reprend les étapes décrites ci-dessus.



Cette démarche lourde et parfois complexe est adaptée à l'automatisation de « gros systèmes d'information ». Pour des informatisations plus modestes, elle peut être perçue comme un carcan, et il convient donc de l'adapter afin de retenir uniquement les concepts et/ou les étapes appropriées aux besoins.

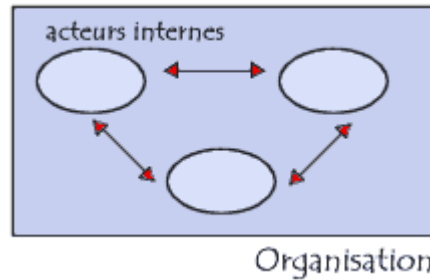
## 1.6 Le modèle conceptuel de la communication :

### 1.6.1 Définition de l'organisation



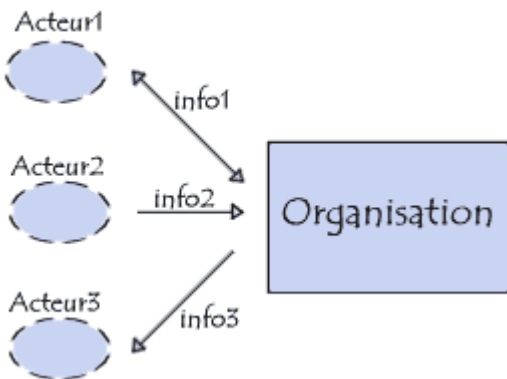
La première étape de ce modèle est d'arriver à isoler le système en le délimitant. Il s'agit donc de définir le système et les éléments externes avec lesquels il échange des flux d'information. Ces éléments extérieurs sont appelés **acteurs externes** (ou partenaires).

La seconde étape consiste à découper l'organisation en entités appelées *acteurs internes* (ou domaines). Lorsque les domaines d'une organisation sont trop importants, ils peuvent être décomposés eux-mêmes en *sous-domaines*.



La dernière étape est l'analyse des flux d'information, c'est-à-dire la définition des *processus*.

### 1.6.2 Diagramme de contexte



Le diagramme de contexte a pour but de représenter les flux d'informations entre l'organisation et les acteurs externes selon une représentation standard dans laquelle chaque objet porte un nom:

l'organisation est représentée par un rectangle

les acteurs externes sont représentés par des ellipses en pointillés

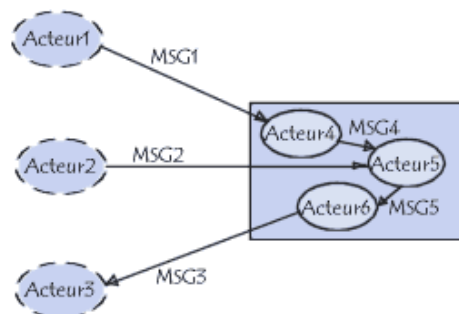
les flux d'information sont représentés par des flèches dont l'orientation désigne le sens du flux d'information

### 1.6.3 Diagramme conceptuel de flux

Ce diagramme (appelé aussi **modèle conceptuel de la communication**) permet de compléter le diagramme de contexte en décomposant l'organisation en une série d'acteurs internes. Dans ce diagramme la représentation standard est la suivante:

Les acteurs internes sont représentés par des ellipses,

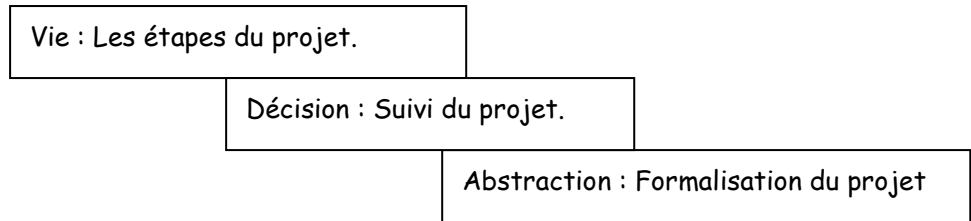
Les messages internes sont représentés par des flèches



## 2 LA DEMARCHE DE LA METHODE MERISE :

### 2.1 Généralités :

#### 2.1.1 3 axes :



#### 2.1.2 Les trois niveaux d'analyse

Le **premier principe** de la méthode consiste à distinguer les différents niveaux successifs d'analyse du système d'information à informatiser.

##### 1- Le niveau conceptuel:

Il exprime le choix de gestion indépendamment des moyens à mettre en oeuvre et de leur organisation

##### 2- Le niveau logique et organisationnel:

Il reflète les choix de moyen et de ressources, et d'organisation de ces ressources, en faisant abstraction de leurs caractères précises

##### 3- Le niveau physique:

Il traduit les choix techniques.

Le **second principe** consiste à séparer les données et les traitements

Le tableau ci-dessous résume les niveaux d'analyse et les modèles associés. Les colonnes "Données" et "Traitements" sont entourées de cercles rouges, et des bulles de dialogue au-dessus indiquent "Année 1" et "Année 2".

Niveau d'analyse	Données	Traitements
Niveau conceptuel	<a href="#">MCD</a>	<a href="#">MCT</a>
Niveau logique	<a href="#">MLD</a>	<a href="#">MOT</a>
Niveau physique	<a href="#">MPD</a>	<a href="#">MPT</a>

A chaque niveau, le produit de l'analyse doit être consigné dans un schéma appelé également modèle.

**MCD**: décrit la signification des données sur lesquels repose le système d'information et les structures.  
(Modèle Conceptuel de Données)

**MCT**: formalise les activités du domaine étudié.  
(Modèle conceptuel des traitements)

**MLD**: fournit une description des données tenant compte des moyens informatiques mis en oeuvre.  
(Modèle Logique des Données)

**MOT**: décrit le fonctionnement du domaine étudié en présentant les ressources mises en oeuvre et leur organisation.  
(Modèle Organisationnel des traitements)

**MPD**: est une description de la base de données ou de l'ensemble des fichiers correspondants aux données gérées par le  
(Modèle physique des données) système d'information.

**MPT**: décrit les spécificités des différents modules de traitement.  
(Modèle physique des traitements)



## 2.1.3 La méthode MERISE

### 2.1 Utilisation empirique de la méthode.

L'utilisation de cette méthode entraîne des conséquences graves. Cela conduit à mettre en oeuvre une base de données mal conçue et comportant de nombreuses redondances inopportunes. Cela rend difficile l'exploitation d'une base de données.

### 2.2 Les logiciels fondés sur MERISE.

Des logiciels ont été développés pour réaliser des MCD et générer automatiquement les tables du modèle relationnel correspondant. Ces logiciels permettent également de réaliser des MCT. Mais ne permettent pas de compenser la méconnaissance de la méthode MERISE.

En effet, ces logiciels se comportent d'abord comme un outil de dessin permettant de tracer un MCD ou un MCT, et il accepte également de dessiner des dessins faux, incomplets, redondants.

L'utilisation d'un tel logiciel n'est donc pas absolument une garantie d'exactitude du modèle élaboré, ni de la base de données générée. D'où, l'importance de bien connaître les différents étages de la modélisation des données pour concevoir un MCD parfaitement exact.

### 2.3 Maîtrise des MCD

Un MCD est un outil de communication entre l'utilisateur de l'informatique et l'informaticien. Pour que l'utilisateur puisse avoir un impact sur ce que leur propose le service informatique en réponse à leur demande, il faut qu'il soit capable de comprendre un MCD et également de le compléter et de le corriger.

## **2.2 L'analyse (ou formalisation) conceptuelle :**

### **2.2.1 Introduction :**

C'est l'étape la plus importante

Elle fixe les choix pour les données et les traitements du SI

Elle n'aborde pas les aspects d'organisation (voir §2.3)

Deux méthodes sont utilisées : MCD (§2.2.2) et MCT (§2.2.3)

### **2.2.2 MCD (Modèle Conceptuel de Données)**

#### **2.2.2.1 Description statique du système d'information**

Le modèle conceptuel des données est une représentation statique du système d'information de l'entreprise qui met en évidence sa sémantique. Il a pour but d'écrire de façon formelle les données qui seront utilisées par le système d'information. Il s'agit donc d'une représentation des données, facilement compréhensible. Cet aspect recouvre les mots qui décrivent le système ainsi que les liens existants entre ces mots. Le formalisme adopté par la méthode Merise pour réaliser cette description est basé sur les concepts « entité-association ».

### La propriété (ou attribut ou rubrique)

#### 2.2.3 1.1) La propriété (ou attribut ou rubrique)

- La propriété est une information élémentaire, c'est-à-dire non déductible d'autres informations, qui présente un intérêt pour le domaine étudié. *Par exemple, si l'on considère le domaine de gestion des commandes d'une société de vente par correspondance, les données : « référence article », « désignation article », « prix unitaire HT », « taux de TVA » sont des propriétés pertinentes pour ce domaine. La donnée « prix unitaire TTC » n'est, d'après la définition, pas une propriété car ses valeurs peuvent être retrouvées à partir des propriétés « prix unitaire HT » et « taux de TVA ».*
- Chaque valeur prise par une propriété est appelée occurrence. *Des occurrences de la rubrique « désignation article » sont par exemple : « râteau », « bêche », « scie », ...*
- Une propriété est dite simple ou encore atomique si chacune des valeurs qu'elle regroupe n'est pas décomposable. *La propriété « Adresse », dont des exemples d'occurrences sont donnés ci-dessous, n'est pas élémentaire car elle peut être décomposée en trois propriétés : la rue, le code postal et la ville.*

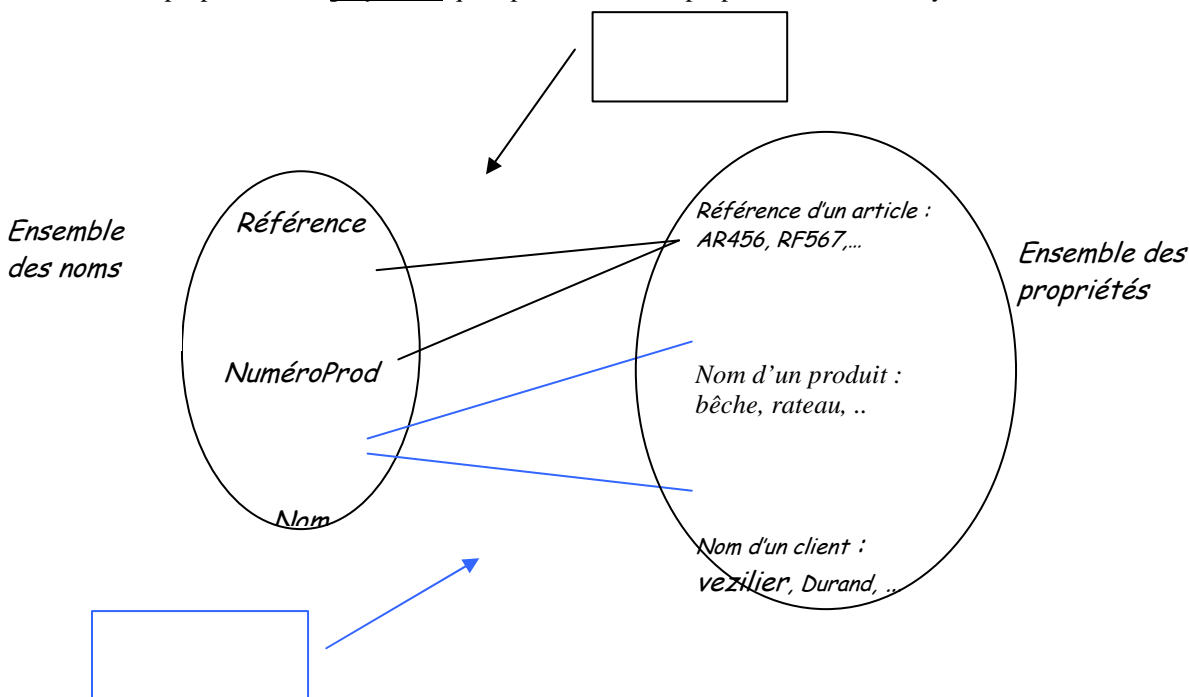
Adresse
310, rue de la gare 59000 Lille
45, avenue de la Plage 593500 St André

La décomposition d'une propriété en propriétés plus simples ne doit pas être systématique et doit surtout tenir compte de son exploitation dans le système. Si cette exploitation est toujours globale, l'atomisation n'est pas nécessaire, dans les autres cas il faut procéder à l'isolement de chacune des composantes de la propriété et donc introduire de nouvelles propriétés.

Propriété	Occurrences
Prénom	Christophe, Laurent, Jean

- Une propriété paramètre est une propriété qui, à un instant donné, contient une seule valeur. *Un des exemples les plus classiques pour illustrer les paramètres est la rubrique « ValeurEuro ».*

Dans le modèle conceptuel des données figurent toutes les propriétés, identifiées par un nom, qui présentent un intérêt pour le domaine à étudier. Ce nom doit être le plus explicite possible : à sa seule lecture on doit donc pouvoir se faire une idée de ce que représente la propriété. En outre, l'identification de chaque propriété consiste à garantir une bijection entre l'ensemble des noms et l'ensemble des propriétés à gérer. On devra donc **exclure** les synonymes qui correspondent à deux noms différents pour identifier la même propriété et les polysèmes qui représentent deux propriétés différentes ayant le même nom.



Enfin, le principe de non-redondance impose que chaque propriété, correctement identifiée, n'apparaisse qu'une seule fois dans le modèle.

## 2.2.4 1.2) L'entité ou individu-type

### 2.2.4.1 a) Définition

Une entité est la représentation d'un élément matériel ou immatériel ayant un rôle dans le système que l'on désire décrire. On appelle **classe d'entité** un ensemble composé d'entités de même type, c'est-à-dire dont la définition est la même. Le classement des entités au sein d'une classe s'appelle *classification* (ou *abstraction*). Une entité est une *instanciation* de la classe. Chaque entité est composée de propriétés, données élémentaires permettant de la décrire.

*Prenons par exemple une Ford fiesta, une Renault Laguna et une Peugeot 306. Il s'agit de 3 entités faisant partie d'une classe d'entité que l'on pourrait appeler voiture. La Ford Fiesta est donc une instanciation de la classe voiture. Chaque entité peut posséder les propriétés couleur, année et modèle.*

Au premier abord, on peut définir l'entité comme étant un regroupement bien pensé, donc sensé, de plusieurs propriétés. Par exemple, on considère l'entité **ARTICLE** qui regroupe les propriétés : *Référence, Désignation et PrixUnitaireHT*. Le droit d'entrée d'une propriété dans une entité est soumis à d'autres facteurs que le bon sens, et ce sont ces facteurs que l'on va étudier.

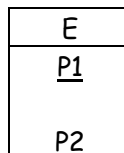


Les classes d'entités sont représentées par un rectangle. Ce rectangle est séparé en deux champs:

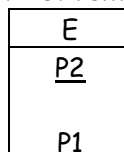
- le champ du haut contient le libellé. Ce libellé est généralement une abbréviation pour une raison de simplification de l'écriture. Il s'agit par contre de vérifier qu'à chaque classe d'entité correspond un et un seul libellé, et réciproquement.
- le champ du bas contient la liste des propriétés de la classe d'entité.

Considérons deux propriétés P1 et P2. La création d'une entité E regroupant ces deux seules propriétés n'est envisageable que si l'une des deux conditions suivantes est satisfaite :

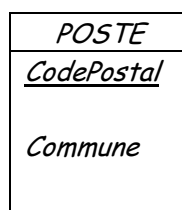
- à toute valeur de la propriété P1 doit correspondre au plus une valeur de la propriété P2. Ce fait traduit l'existence d'une dépendance fonctionnelle monovaluée entre P1 et P2 notée :  $P1 \rightarrow P2$ . On dit encore que P1 détermine P2. P1 est alors rubrique identifiante de l'entité E. La représentation graphique de l'entité E a la forme suivante :



- ou à toute valeur de la rubrique P2 doit correspondre au plus une valeur de la rubrique P1. P2 est alors en dépendance fonctionnelle avec P1 et l'entité E doit être représentée ainsi :



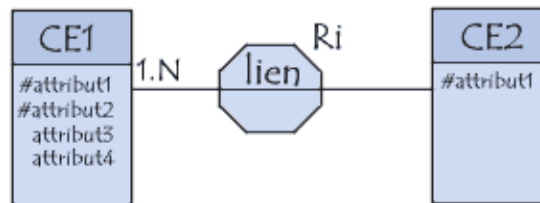
*L'entité suivante, qui peut être considérée comme un regroupement sensé, n'est pas correcte car il n'y a pas dépendance fonctionnelle entre la rubrique « CodePostal » et la rubrique « Commune ». Ainsi au code postal « 16600 » correspond plusieurs communes : « Mornac », « Magnac », ..*



## 2.2.4.2 b) Les identifiants

Un identifiant est un ensemble de propriétés (une ou plusieurs) permettant de désigner une et une seule entité. La définition originale est la suivante: **L'identifiant est une propriété particulière d'un objet telle qu'il n'existe pas deux occurrences de cet objet pour lesquelles cette propriété pourrait prendre une même valeur.**

Les attributs d'une classe d'entité permettant de désigner de façon unique chaque instance de cette entité sont appelé identifiant absolu. Le modèle conceptuel des données propose de souligner les identifiants (parfois de les faire précéder d'un #).



Ainsi, chaque classe d'entité doit posséder au moins un attribut identifiant, et l'ensemble de ses attributs identifiants doivent être renseignés à la création de l'entité.

## 2.2.4.3 c) Occurrence d'entité ou individu

D'après la définition d'une entité, on sait que la connaissance d'une valeur de la rubrique identifiante détermine la connaissance des valeurs des autres rubriques de l'entité. L'ensemble de ces valeurs est appelé occurrence d'entité. Le tableau suivant présente des exemples d'occurrences de l'entité *ARTICLE*.

<i>ARTICLE</i>		
<u>Référence</u>	<i>134ER</i>	<i>354TY</i>
Désignation	<i>Râteau</i>	<i>Bêche</i>
PrixUnitaireHT	<i>150 F</i>	<i>68,50 F</i>
	<i>452GT</i>	
	<i>Scie</i>	
	<i>45 F</i>	

## d) Notion de dépendance fonctionnelle directe

Considérons l'entité suivante et quelques une de ses occurrences :

ARTICLE	134ER	354TY	452GT
Référence	Râteau	Bêche	Scie
Désignation	150 F	68,50 F	45F
PrixUnitaireHT	A	A	B
NoCatégorie	Jardinage	Jardinage	Bricolage
LibelléCatégorie			

Cette entité est juste mais elle implique une redondance d'information relative à la catégorie. L'association entre le numéro de la catégorie et son libellé est en effet répétée dans chaque occurrence de l'entité ARTICLE.

Pour supprimer de telles redondances, on devra veiller à ce que toute dépendance fonctionnelle entre la propriété identifiante de l'entité et une propriété non identifiante de l'entité soit directe. Une **dépendance fonctionnelle monovaluée  $x \rightarrow y$  est directe** s'il n'existe pas de propriété  $z$  telle que :  $x \rightarrow z$  et  $z \rightarrow y$ .

Dans l'exemple précédent la dépendance fonctionnelle Référence  $\rightarrow$  LibelléCatégorie n'est pas directe car il existe la propriété NoCatégorie telle que :

Référence  $\rightarrow$  NoCatégorie et NoCatégorie  $\rightarrow$  LibelléCatégorie

### 2.2.4.4 Exercice n°1 (Acquis : Propriété et Entité)

La société nord-Hebdo consacre l'essentiel de son activité à l'édition et à la distribution d'un journal spécialisé dans les petites annonces et la publicité dans le nord de la France. La parution du journal est hebdomadaire et sa distribution est assurée uniquement dans les départements du nord et Pas de C.

La tarification d'une annonce est fournie ci-dessous :

➤ Première semaine de parution : tarif pour 5 lignes au plus :

Rubrique de l'annonce	Prix
Emploi	50 F
Bourse aux affaires	40 F
Tout ce qui roule	55 F
Immobilier	55 F
Contacts	75 F
Loisirs	50 F

➤ Options

Prix de la ligne supplémentaire : 50 F

Domiciliation (pour préserver l'anonymat du client) : 80 F

Semaines supplémentaires : des réductions sont accordées selon le tableau ci-dessous :

Période	Pourcentage de réduction *
2 <sup>ème</sup> semaine	20%
3 <sup>ème</sup> semaine et suivantes	40 %

\*Ces réductions sont applicables au prix de base de la première semaine.

Le tableau ci-dessous répertorie un ensemble de données qui se rapporte à la gestion des annonces (La liste est triée sur le nom de la donnée).

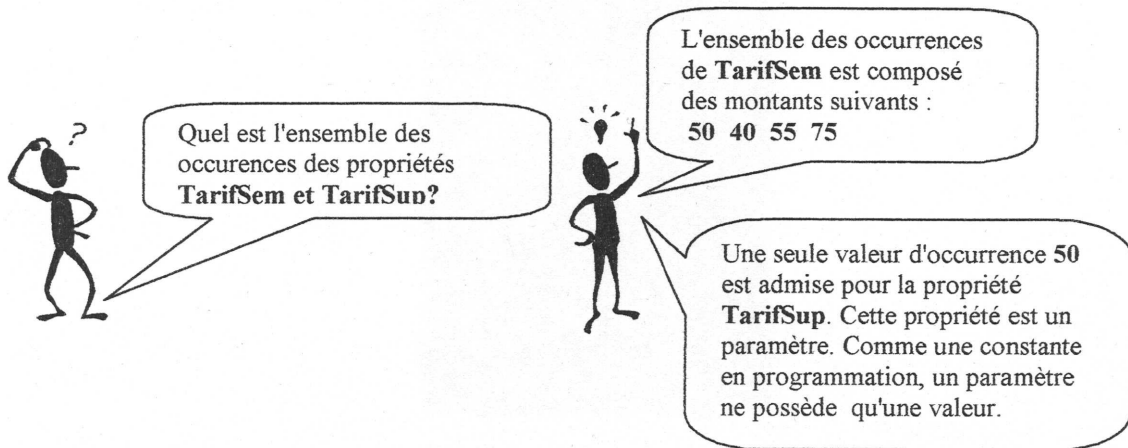
Nom	Signification
CP	Code postal du client qui dépose l'annonce
DateRedac	Date de dépôt de l'annonce
Dom	Domiciliation de l'annonce déposée(oui/non)
NbLignes	Nombre de lignes de l'annonce déposée
NbSem	Nombre de semaines de parution de l'annonce déposée
Nom	Nom du client qui dépose l'annonce
NumAnn	Numéro qui identifie chaque annonce déposée
NumCli	Numéro qui identifie chaque client qui dépose une annonce
Prénom	Prénom du client qui dépose l'annonce
Prix	Prix de l'annonce déposée
Rub	Rubrique de l'annonce déposée : Emploi, Contacts
Rue	Première partie de l'adresse du client qui dépose une annonce
TarifDom	Tarif de la domiciliation
TarifSup	Tarif de la ligne supplémentaire
TarifPrem	Tarif de la première semaine
Texte	Texte de l'annonce
Ville	Ville de l'adresse du client qui dépose l'annonce

1. Indiquer, parmi les données ci-dessus, celles qui peuvent être qualifiées de propriétés.

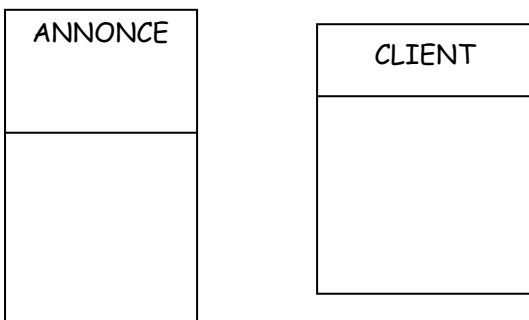
Parmi la liste des données, on recherche les informations non déductibles d'autres informations et qui ont un intérêt pour le domaine étudié. Or, parmi cette liste, **deux** données seulement **ne peuvent pas être considérées comme des propriétés**:

- NbLignes: donnée qui peut être déduite de la propriété Texte et éventuellement d'une propriété paramètre fournissant le nombre de caractères par ligne,
- Prix : donnée calculée à partir des propriétés TarifDom, TarifSup, TarifPrem.

2. En vous basant sur les éléments de tarification, fournir l'ensemble des occurrences des propriétés : TarifPrem et TarifSup. Parmi ces deux propriétés laquelle est une propriété paramètre ?



3. On considère l'ébauche du modèle conceptuel de données suivante :



3.1 On considère l'événement suivant :

Celibat Claire qui habite : 72, Avenue de la gare 59000 LILLE dépose le 14/02/2003 l'annonce ci-après :

Annonce n° 465

F seule ch compagnon

Pas Sérieux s'abstenir

Ecrire sous référence 52369 au journal

(Nombre de semaines de parution : 3)

En limitant le système d'information à cette annonce, fournir les occurrences de l'entité CLIENT et ANNONCE.

ANNONCE
<u>NumAnn</u>
Texte
NbLignes
Dom
NbSem
DateRédac
Rub
TarifSem

465
F Seule ch compagnon
Sérieux s'abstenir
2
OUI
3
14/02/2001
Contact
75F

CLIENT
<u>NumCli</u>
Nom
Prénom
Rue
CP
Ville

52369
Figeac
Claire
72 avenue de la gare
05000
NICE

3.2 Les dépendances fonctionnelles issues de la conception de l'entité ANNONCE sont-elles toutes directes ? justifier votre réponse.

La dépendance fonctionnelle NumAnnn  $\rightarrow$  TarifPrem n'est pas une dépendance fonctionnelle directe. Elle peut, en effet, être retrouvée par transitivité grâce aux deux dépendances fonctionnelles élémentaires suivantes:

NumAnn  $\rightarrow$  Rub

Rub  $\rightarrow$  TarifSem

Pour corriger ce MCD, il faut supprimer la dépendance fonctionnelle directe tout en maintenant les deux dépendances fonctionnelles NumAnn  $\rightarrow$  Rub et Rub  $\rightarrow$  TarifPrem.

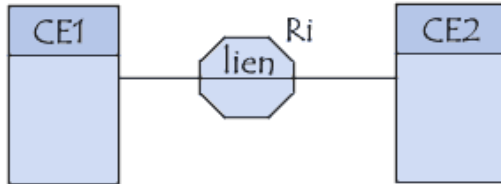
L'idée est de créer une nouvelle entité RUBRIQUE dont Rub est la propriété identifiante. Pour garder la dépendance fonctionnelle NumAnn  $\rightarrow$  Rub il faut relier les entités Rubrique et Annonce par des associations.



## 2.2.5 1.3) L'association (ou relation-type)

### 2.2.5.1 a) Définition

Une association (appelée aussi parfois *relation*) est un lien sémantique entre plusieurs entités. Une classe de relation contient donc toutes les relations de même type (qui relient donc des entités appartenant à des mêmes classes d'entité).

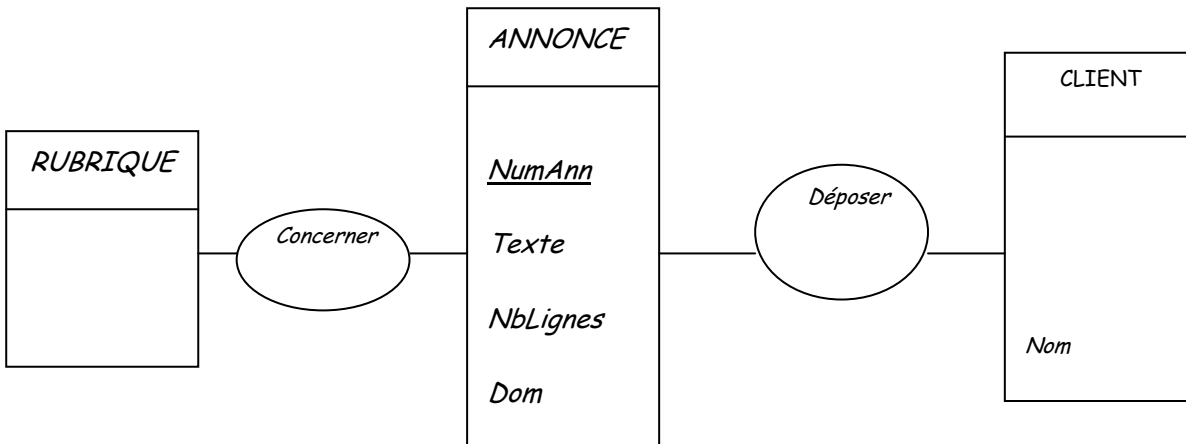


Une classe de relation peut lier plus de deux classes d'entité. Voici les dénominations des classes de relation selon le nombre d'intervenants:

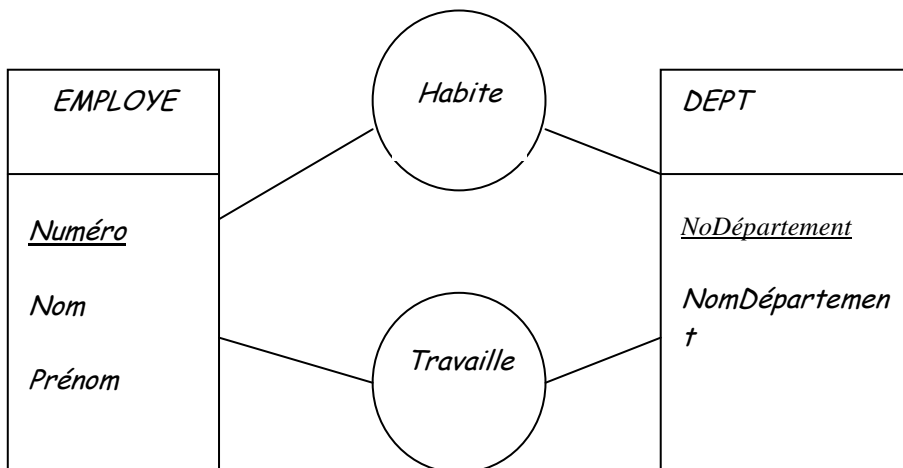
- une classe de relation **récursive** (ou *réflexive*) relie la même classe d'entité
- une classe de relation **binnaire** relie deux classes d'entité
- une classe de relation **ternaire** relie trois classes d'entité

Une classe de relation **n-aire** relie n classes d'entité Les classes de relations sont représentées par des hexagones (parfois des ellipses) dont l'intitulé décrit le type de relation qui relie les classes d'entité (généralement un verbe). On définit pour chaque classe de relation un identificateur de la forme  $R_i$  permettant de désigner de façon unique la classe de relation à laquelle il est associé. On peut éventuellement ajouter des propriétés aux classes de relation.

Par exemple, dans le modèle conceptuel de données relatif à la gestion des annonces on introduit les deux associations « Déposer » et « Concerner » afin d'exprimer les réalités suivantes : un client dépose une annonce et une annonce concerne une rubrique.



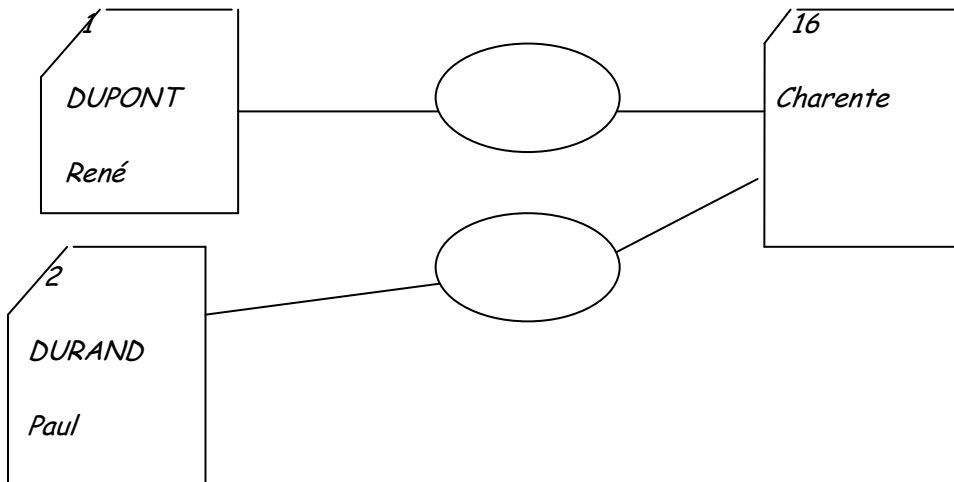
Il peut y avoir, entre deux mêmes entités, plusieurs associations qui représentent chacune des réalités différentes. Dans le modèle conceptuel des données ci-dessous, l'association Habite indique le département dans lequel l'employé habite et l'association Travaille fournit le département dans lequel il travaille. Pour certains employés ces deux départements peuvent être identiques.



L'ensemble des entités participant à une association est appelé collection de cette association. La collection de l'association HABITE est formée des entités : EMPLOYE et DEPT.

### 2.2.5.2 b) Occurrence d'association

Une occurrence d'association est un lien particulier qui relie deux occurrences d'entités. Le schéma ci-dessous présente deux exemples d'occurrences de l'association « Habite ».



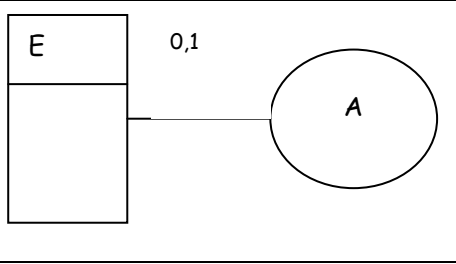
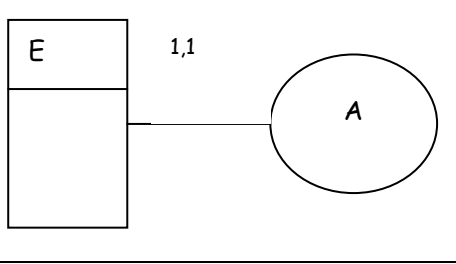
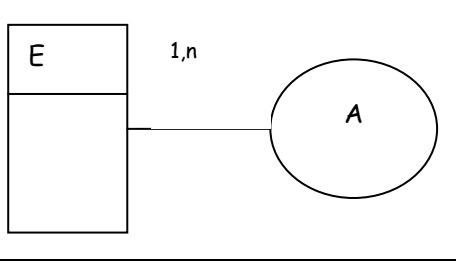
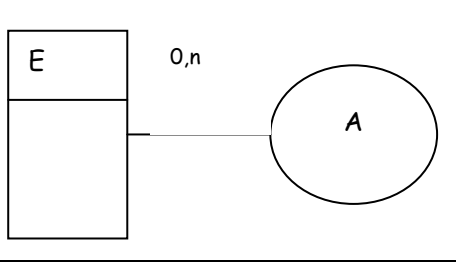
**Remarque :** certains auteurs définissent l'identifiant d'une association comme étant la concaténation des identifiants des entités qui participent à l'association.

### 2.2.5.3 c) Cardinalité

Les cardinalités permettent de caractériser le lien qui existe entre une entité et la relation à laquelle elle est reliée. La cardinalité d'une relation est composé d'un couple comportant une borne maximale et une borne minimale, intervalle dans lequel la cardinalité d'une entité peut prendre sa valeur:

- la borne minimale (généralement 0 ou 1) décrit le nombre minimum de fois qu'une entité peut participer à une relation
- la borne maximale (généralement 1 ou n) décrit le nombre maximum de fois qu'une entité peut participer à une relation

Un couple de cardinalités placé entre une entité E et une association A représente le nombre minimal et maximal d'occurrences de l'association A qui peuvent être « ancrées » à une occurrence de l'association E. Le tableau ci-après récapitule les valeurs que peut prendre ce couple.

	<p>Pour chaque occurrence de E, le modèle admet :</p> <ul style="list-style-type: none"> <li>- soit l'absence de lien</li> <li>- soit la présence d'un seul lien</li> </ul>
	<p>Pour chaque occurrence de E le modèle admet la présence d'un et un seul lien</p>
	<p>Pour chaque occurrence de E le modèle admet la présence d'un seul ou de plusieurs liens</p>
	<p>Pour chaque occurrence de E le modèle admet :</p> <ul style="list-style-type: none"> <li>- soit l'absence de lien</li> <li>- soit la présence de plusieurs liens</li> </ul>

**Remarque :** dans certaines situations, la lettre n peut être remplacée par une valeur.

## 2.2.5.4 Exercice n°2 (Acquis : propriété, entité et association)

Le système d'information étudié concerne l'activité de gestion des locations saisonnières d'une agence immobilière. Une analyse de l'existant a permis de dégager les entités suivantes :

Entité	Objectif	Propriétés
PROPRIETAIRE	Regroupe toutes les informations relatives aux propriétaires d'appartements	NumPropriétaire Nom Prénom Adresse1 Adresse2 CodePostal Ville NumTel1 NumTel2 E-mail Cacumulé
APPARTEMENT	Regroupe toutes les informations des appartements meublés mis à la location	NumLocation Catégorie : 1, 2, ou 3 étoiles Type : T2, T3, T4 NbPersonnes AdresseLocation Photo Equipements
LOCATAIRE	Regroupe toutes les informations sur les locataires qui ont effectué au moins une location par l'intermédiaire de l'agence	NumLocataire NomLocataire PrénomLocataire Adresse1Locataire Adresse2Locataire CodePostalLocataire VilleLocataire NumTel1Locataire NumTel2Locataire E-mailLocataire
CONTRAT	Regroupe toutes les informations relatives à une location qui va avoir lieu ou qui a actuellement lieu. Une location s'étend éventuellement sur plusieurs semaines consécutives.	NumContrat Etat : réservé, confirmé, soldé DateCréation DateDébut DateFin
TARIF	Regroupe les informations liées à la tarification	CodeTarif PrixSemHS (prix semaine haute saison) PrixSemBS (prix semaine basse saison)

1. Pourquoi l'information *CAcumulé* de l'entité PROPRIETAIRE est-elle une propriété ?

Bien que pouvant être une donnée calculée par le système d'information, il est préférable d'avoir le chiffre d'affaire cumulé en propriété de manière à pouvoir accéder plus rapidement à l'information (allègement des traitements). C'est une "dénormalisation".

2. La propriété Equipements est destinée à décrire les principaux équipements de l'appartement : téléviseur, lave-vaisselle, ... Quels sont les inconvénients liés à une telle propriété ?

Le fait de regrouper toutes les caractéristiques des équipements dans une même propriété ne permettra pas de faire des traitements précis sur ces derniers. Par exemple, il sera impossible de lister tous les appartements de type T2 qui possède un lave-vaisselle.

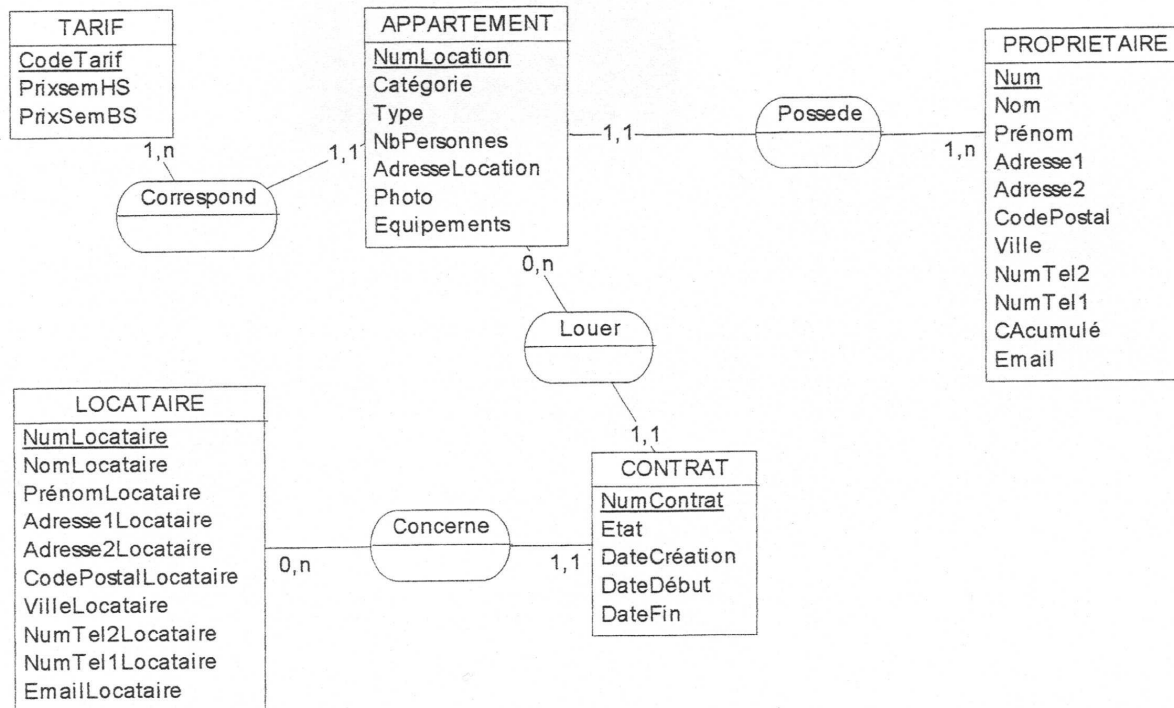
L'isolement des différents composants peut être réalisé de deux manières:

- Introduire dans l'entité APPARTEMENT des propriétés booléennes telles que Téléviseur, Lave-vaisselle etc.
- Créer une entité TYPE-EQUIPEMENT et mettre en place une association (m-n) entre l'entité APPARTEMENT et l'entité TYPE-EQUIPEMENT.

3. Présenter le modèle conceptuel des données décrivant ce système d'information en tenant compte des règles de gestion suivantes :

- La notion de co-propriété ne doit pas être prise en compte ce qui revient à dire que tout appartement appartient à un et un seul propriétaire.
- A tout appartement correspond un code tarif

Seules les noms des entités figureront sur le modèle.



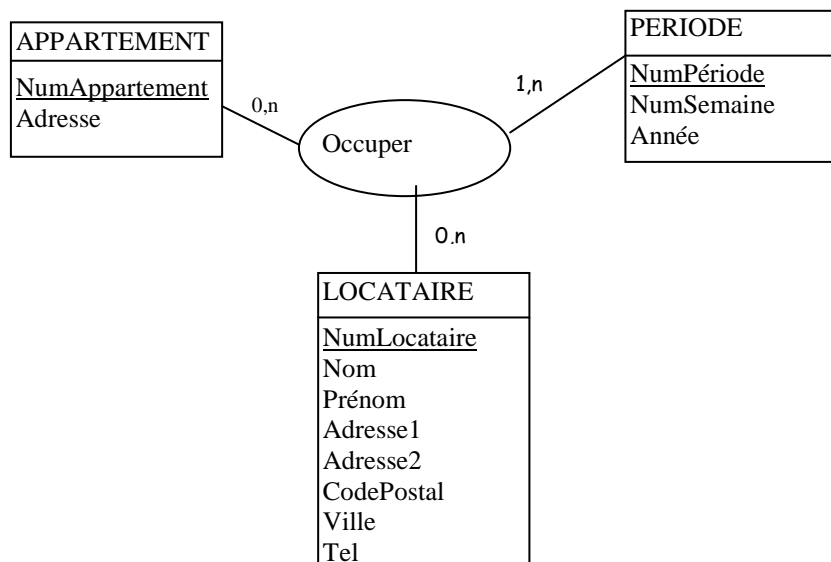
4. On restreint le domaine étudié à la gestion des locations des appartements possédés par M. X. Les entités recensées sont données ci-dessous :

Entité	Objectif	Propriétés
APPARTEMENT	Regroupe toutes les informations relatives aux appartements de M. X	<u>NumAppartement</u> Adresse
PERIODE	Cette entité admet une occurrence par semaine réservée ou occupée	<u>NumPériode</u> NumSemaine Année
LOCATAIRE	Regroupe toutes les informations sur le locataire	NumLocataire Nom Prénom Adresse1 Adresse2 CodePostal Ville Tel

Pour une semaine donnée, un appartement de M. X peut être :

- soit réservé ou occupé par un locataire
- soit libre
- soit indisponible (ce cas correspond à l'occupation de l'appartement par M.X)

Discuter la proposition de modélisation suivante qui est destinée à représenter l'occupation des appartements de M. X :

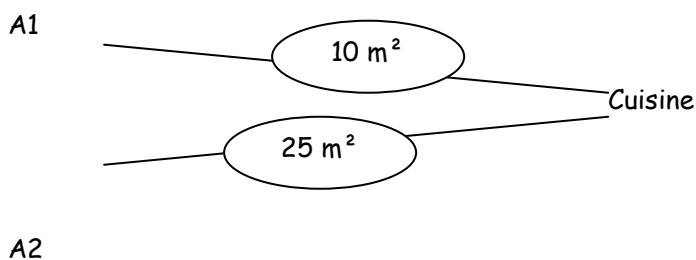
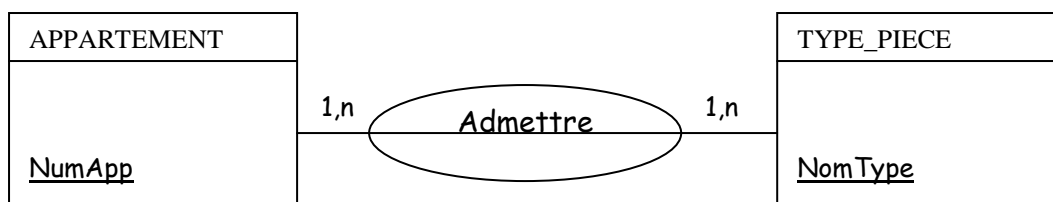


Cette représentation n'est pas satisfaisante car elle admet que le locataire 2205 occupe l'appartement n°1 pour la période 3 et que le locataire 1903 occupe également l'appartement n°1 pour la période 3. Le système d'information accepte qu'un appartement soit loué pour la même période à deux locataires différents.

Il faut pouvoir représenter le fait que pour un couple d'occurrence APPARTEMENT-PERIODE, on n'admette qu'une et une seule occurrence de LOCATAIRE.

5. On souhaite décrire pour chaque appartement les différentes pièces qui le composent ainsi que leur superficie. Par exemple : l'appartement n° 345 possède une kitchenette de 4 m<sup>2</sup>, une salle de bains de 4 m<sup>2</sup>, un séjour de 20 m<sup>2</sup> et une terrasse de 5m<sup>2</sup>.

Enrichir le modèle conceptuel afin de représenter une telle réalité



### 2.2.5.5 d) Caractéristiques d'une association

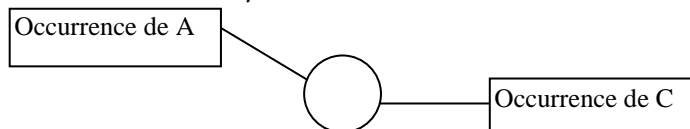
#### ➤ La dimension d'une association

La dimension d'une association indique le nombre d'entités participant à l'association. Les dimensions les plus courantes sont 2 (association binaire) et 3 (association ternaire) :

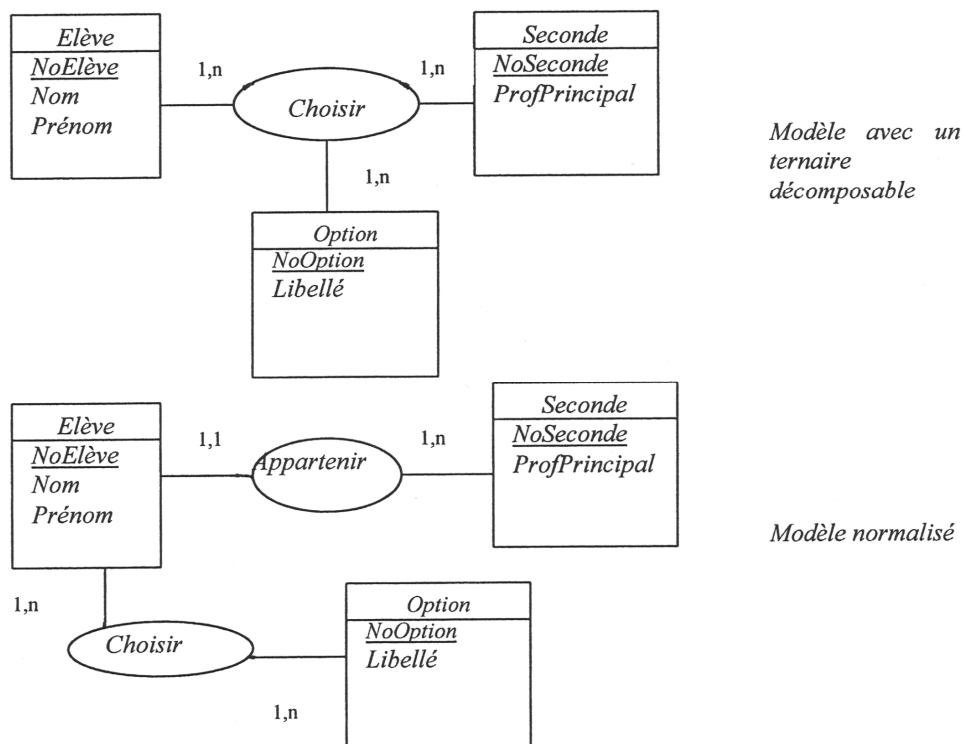
- L'association binaire exprime la présence de liens sémantiques entre les occurrences d'une entité A et les occurrences d'une entité B. L'association « Habite » entre les EMPLOYE et HABITE est binaire.
- L'association ternaire exprime la présence de liens sémantiques entre les occurrences de 3 entités.

#### Remarques :

- Toute occurrence d'une association de dimension n doit être reliée à n occurrences d'entités. Par exemple, pour une association ternaire dans laquelle participent trois entités « A », « B » et « C », toute occurrence doit être reliées à 3 occurrences des entités respectives A, B et C. On ne peut donc pas avoir une occurrence à 2 pattes de la forme ci-dessous.



- L'opération de décomposition consiste à éclater une relation de dimension n en une ou plusieurs associations de dimension moindre sans perte de sémantique. Dans l'exemple ci-dessous, qui se rapporte à la gestion des options en classe de seconde, l'association ternaire du premier modèle doit être éclatée en deux associations binaires afin d'aboutir au second modèle.



Modèle avec un ternaire décomposable

Modèle normalisé

Les associations figurant dans le modèle conceptuel des données devront être non décomposables. Pour ce faire on recherchera en priorité les associations binaires puis les ternaires.

#### ➤ La fonctionnalité d'une association binaire

Les différents types de fonctionnalités d'une association binaire A définie entre deux entités E1 et E2 sont les suivants :

- un à un (1 - 1) caractérisé par la cardinalité **maximum** égale à 1 sur les deux segments « E1 - A » et « E2 - A »,
- un à plusieurs (1 - n) caractérisé par la cardinalité **maximum** égale à 1 sur l'un des segments et la cardinalité maximum égale à n sur l'autre,
- plusieurs à plusieurs (m-n) caractérisé par la cardinalité **maximum** égale à n sur les deux segments « E1 - A » et « E2 - A ».

#### ➤ Les notions de partialité et de totalité

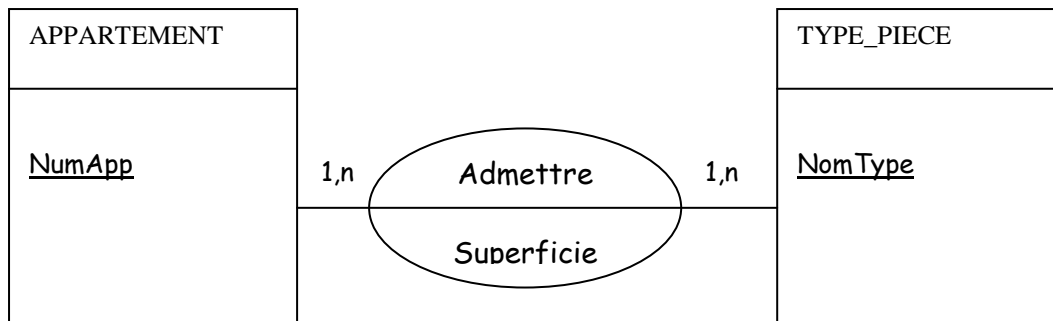
Une association binaire A définie entre les entités E1 et E2 est totale si elle est caractérisée par la cardinalité minimale égale à un sur les deux segments « E1 - A » et « E2 - A ».

Une association binaire A définie entre les entités E1 et E2 est partielle si elle est caractérisée par la cardinalité minimum égale à 0 sur l'un des segments.



## 2.2.5.6 e) L'association porteuse

Les propriétés qui dépendent fonctionnellement de plusieurs identifiants d'entités sont portées par les associations entre ces entités. C'est une dépendance fonctionnelle multi-attributs au niveau de la source. Dans le système d'information de l'exercice n° 2 la propriété *Superficie* est portée par l'association qui relie l'entité *APPARTEMENT* à l'entité *TYPE\_PIECE* car pour un appartement et une pièce on a une et une seule superficie. Par exemple pour l'appartement 134 et la pièce de type kitchenette on a la superficie : 3 m<sup>2</sup>.



Le droit d'entrée d'une propriété P dans une association reliant n entités est donc soumis à l'existence de la dépendance fonctionnelle suivante :  $I_1, \dots, I_n \rightarrow R$

où  $I_1, I_2, \dots$  représentent l'identifiant de chacune des entités qui participent à l'association.

Pour éviter toute redondance, on s'assurera en outre que la dépendance fonctionnelle est élémentaire. Les associations porteuses sont donc toujours de type (m-n). On dit qu'une propriété est en dépendance fonctionnelle élémentaire avec une liste de rubriques LR :

- si elle est fonctionnellement dépendante de LR,
- si elle n'est pas fonctionnellement dépendante d'une sous-liste de LR.

La dépendance fonctionnelle suivante :  $NoAppart, NomType, NumPropriétaire \rightarrow Superficie$  n'est pas élémentaire car il existe la sous-liste  $NumAppart, NomType$  telle que :

$NumAppart, NomType \rightarrow Superficie$ .

## 2.2.5.7 Auto-évaluation n° 1:

Pour faciliter les contacts entre ses différents employés, la société X a confié à son service informatique la réalisation d'un annuaire électronique. L'une des spécifications de cette application précise qu'elle devra être accessible à l'ensemble du personnel par l'intermédiaire de l'INTRANET de l'entreprise.

Un extrait de l'entretien avec le demandeur de l'application M. D est reproduit ci-dessous :

**L'analyste** : Comment sont identifiés les salariés de la société ?

**M. D** : Chaque employé de la société est identifié par un numéro et appartient à un service caractérisé par un nom. Comme exemples je peux vous citer le service comptabilité ou le service production.

**L'analyste** : Peut-il y avoir deux services qui portent le même nom ?

**M. D** : Non.

**L'analyste** : Quelles sont les données relatives à un salarié que vous souhaitez déposer sur l'annuaire

**M.D** : son nom, son prénom, ses coordonnées téléphonique et INTERNET ainsi que le service et la section auquel il est rattaché. Je précise que certains services sont découpés en sections, elles aussi identifiées par un nom. Par exemple, le service informatique comprend les sections études et production. Le service comptabilité n'admet pas de section.

**L'analyste** : Tout salarié a-t-il un poste téléphonique ?

**M.D** : non, certains employés n'ont pas encore de poste téléphonique. Pour les autres ils en ont un et un seul. C'est un numéro interne composé de 4 chiffres. Le mien est par exemple 48 14.

**L'analyste** : Je suppose que tous les salariés n'ont pas une adresse électronique.

**M.D** : Sur notre INTRANET tous les employés ont une adresse professionnelle qui leur permet d'envoyer ou de recevoir du courrier électronique relatif à leur activité.

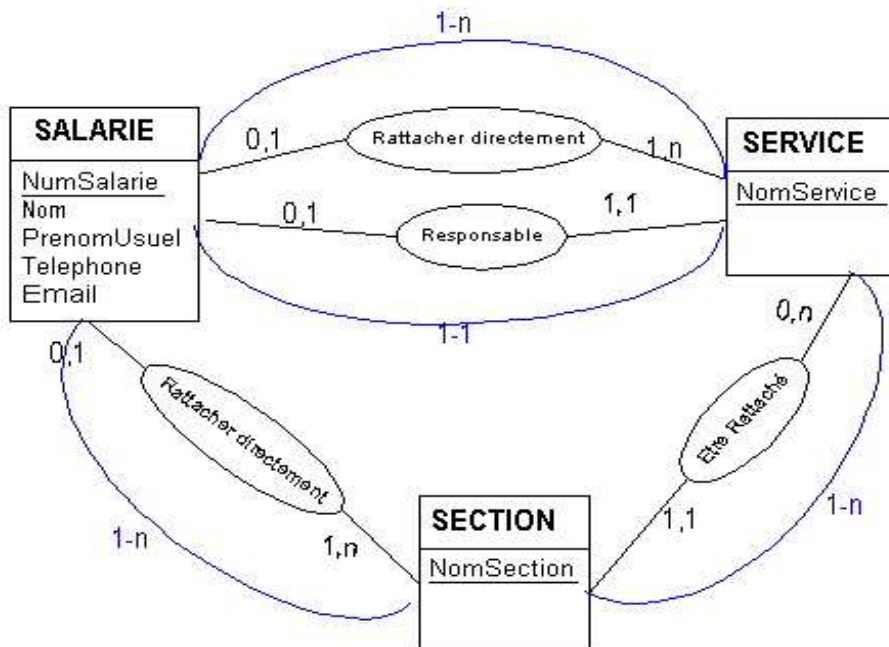
**L'analyste** : Revenons sur les sections. Une section peut-elle concerner plusieurs services ?

**M. D** : Non, une section concerne un et un seul service.

**L'analyste** : En terme de traitements quelles sont vos attentes ?

**M.D** : Ma première attente serait bien évidemment de retrouver très rapidement les coordonnées d'un employé. Dans le cas où l'employé n'a pas de numéro de téléphone, il serait souhaitable d'afficher le nom ainsi que le numéro de téléphone du responsable du service auquel il appartient. Je précise qu'il y a un seul responsable par service.

1. Proposer un modèle conceptuel des données pour la conception de cette application
2. Indiquer pour chacune des associations son type : 1 - 1, 1-n ou m-n



Notons qu'une association est partielle si une des cardinalité est à 0. Donc ici, elles sont toutes partielles.

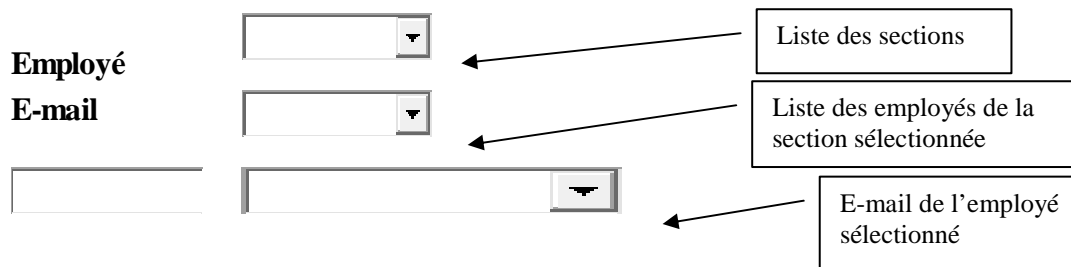
### 3. Lister toutes les incohérences acceptées par votre MCD

- Un salarié peut n'appartenir à rien (ni section, ni service),
- Un salarié peut être responsable d'un autre service que celui dont il fait partie.

Il faut donc mettre en place des contraintes applicatives:

- Un responsable d'un service doit être "Rattaché directement" au même service,
- Tout salarié est rattaché soit à une section, soit à un service.

### 4. Faire apparaître sur le MCD le chemin emprunté lors de la mise œuvre du traitement suivant :

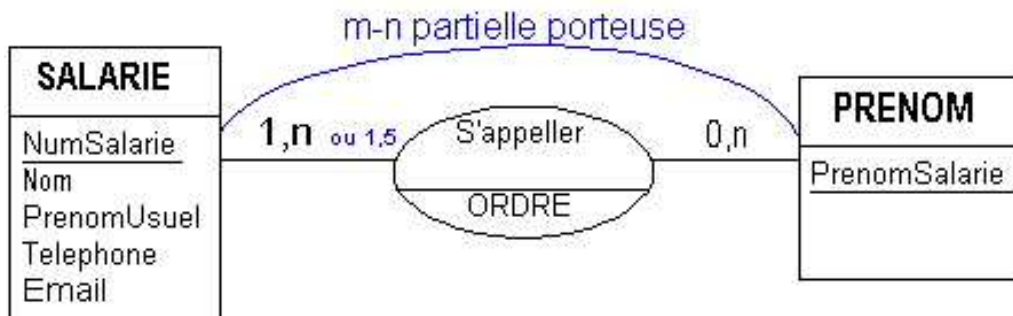


Section: On prend toutes les occurrences de l'entité SECTION,

Employé: Cardinalité 1,n pour avoir les occurrences de SALARIE concernées,

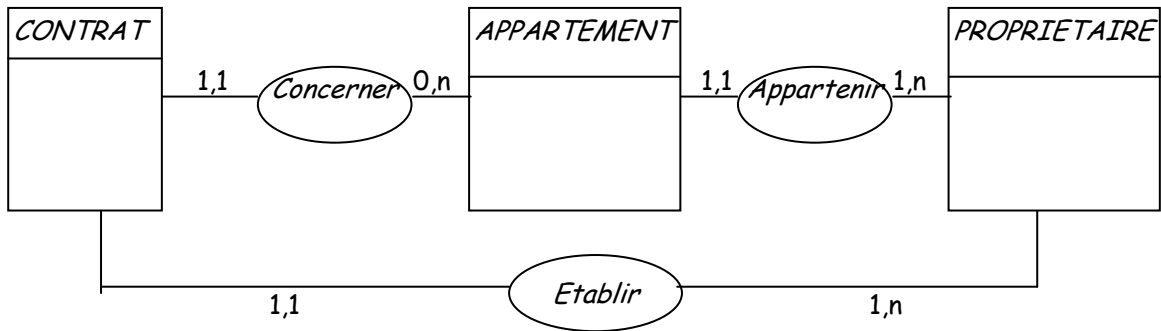
E-mail: Cardinalité 0,1 pour avoir la propriété du SALARIE concerné.

### 5. On souhaite modifier la modélisation de façon à permettre la mémorisation de tous les prénoms d'un employé. Certains, aux parents forts imaginatifs, admettent jusqu'à 5 prénoms ! Proposer une solution qui permet l'enregistrement non seulement des prénoms d'un employé mais aussi de leur ordre dans l'état civil.



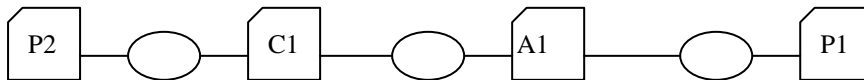
## 2.2.5.8 f) Les associations transitives

Considérons le modèle suivant :



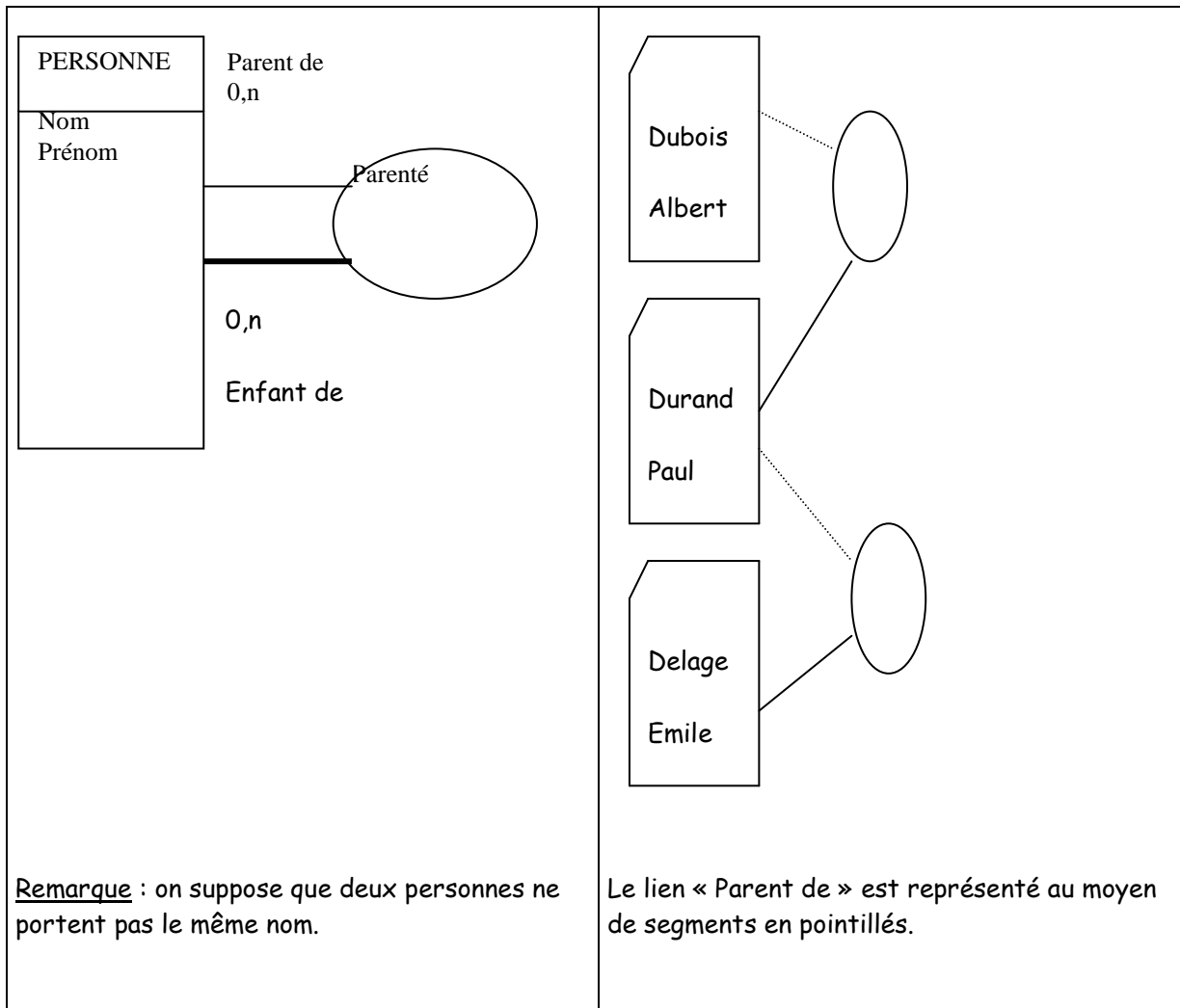
L'association binaire qui relie l'entité « *CONTRAT* » et l'entité « *PROPRIETAIRE* » doit être ôtée du modèle car on peut retrouver le propriétaire à partir des associations « *Concerner* » et « *Appartenir* ». Il s'agit d'une association transitive. Seules les dépendances fonctionnelles directes entre identifiants d'entités devront donner lieu à des association binaires de type (1-n).

On veillera à supprimer les dépendances transitives pour ne pas surcharger le MCD car on risque de faire des incohérences comme:



## 2.2.5.9 g) Les associations réflexives

L'association réflexive est une association binaire qui relie une entité à elle-même. Une occurrence de l'association établit donc un lien entre une occurrence de l'entité et une autre occurrence de cette même entité. Dans le cas d'une association non symétrique, on doit faire porter le rôle sur chacun des segments, *comme l'illustre l'exemple suivant*.



## 2.2.5.10 Exercice n°3 (Acquis : Concepts de base uniquement)

On considère l'univers du discours suivant :

« Le personnel du service informatique de la MAAF peut être réparti en deux catégories :

- le personnel interne qui regroupe les employés rémunérés par la MAAF,
- le personnel extérieur qui englobe des salariés de différentes SSII

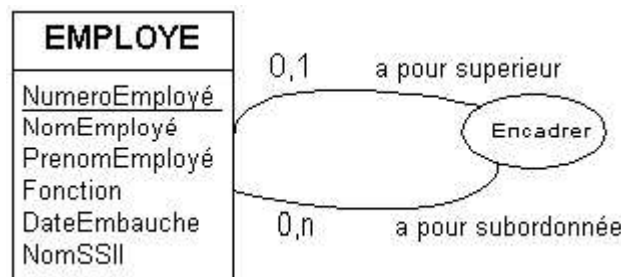
Excepté le directeur du service, chaque employé (extérieur ou interne) est encadré par un supérieur hiérarchique qui est obligatoirement un salarié interne. »

Relativement à la gestion du personnel du service informatique, on retient la liste des données suivantes :

NuméroEmployé	numéro qui identifie chaque employé travaillant au service informatique
NomEmployé	nom de l'employé
PrénomEmployé	prénom de l'employé
Fonction	fonction occupée par l'employé : Analyste-programmeur, chef de projet, ...
DateEmbauche	date de l'embauche. Propriété uniquement définie pour les salariés internes
NomSSII	nom de la société de services. Cette propriété est uniquement définie pour les salariés extérieurs

Proposer un modèle conceptuel de données décrivant cette réalité et permettant de répondre à des requêtes telles que :

- Quel est le nom du supérieur hiérarchique de M.X?
- Quelle est la liste des salariés internes?
- Quelle est la liste des employés ayant une fonction d'encadrement?



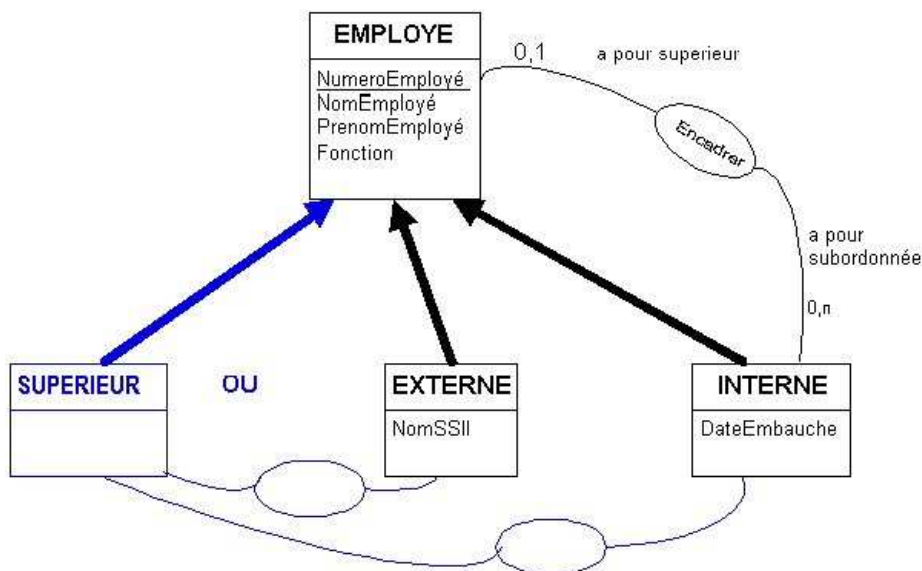
Contrainte applicative: Un employé doit avoir un supérieur "interne".

Contrainte d'exclusion: sur DateEmbauche et NomSSII.

Niveau sémantique: la date d'embauche n'a pas de sens pour un employé externe (idem pour NomSSII et un interne).

Considération physique: optimisation mémoire si NomSSII vide.

**Solution par l'utilisation de Merise 2:**



**2.2.5.11 Auto-évaluation n° 2:**

Dans le cadre de l'automatisation du suivi annuel de formation du personnel de la société x, le responsable des ressources humaines a défini pour chaque poste de l'entreprise, les compétences requises. Par exemple, le poste «infographiste » requiert les compétences: créativité, connaissances techniques et aptitudes relationnelles. Outre cette définition de poste, chaque salarié réalise en fin d'année un bilan de compétences destiné à connaître pour chaque compétence exigée par son poste son niveau actuel. Le tableau suivant récapitule les compétences du salarié n°1325.

N° salarié: 1325	Poste: infographiste		
Nom: Dubois	Prénom: Michel		
Intitulé des compétences	Niveau actuel		
	A	B	C
Connaissances techniques	X		
Créativité		X	
Aptitudes relationnelles		X	

Les niveaux de compétence sont codés ainsi:

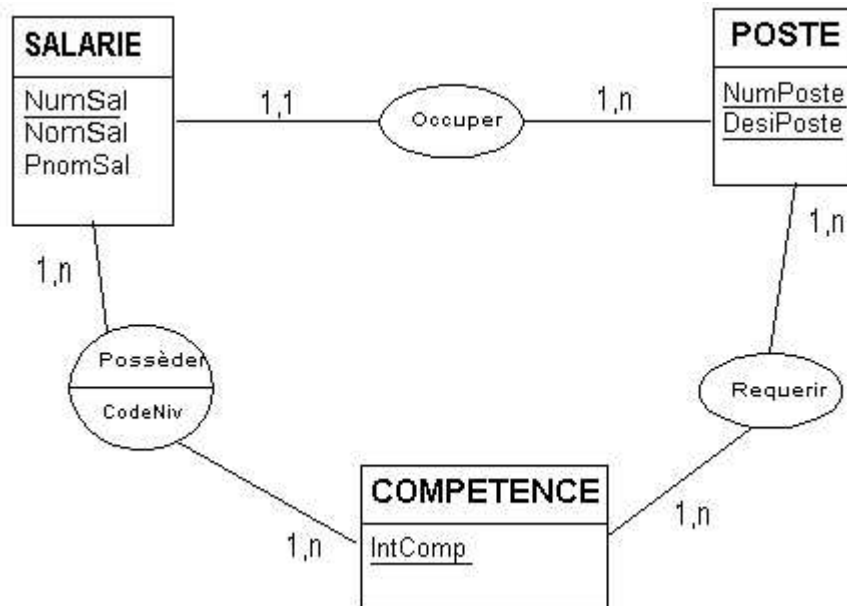
A : compétence confirmée

B : compétence à renforcer

C : compétence en cours d'acquisition

1. Proposer un modèle conceptuel des données permettant de structurer les propriétés figurant dans le tableau suivant:

Nom propriété	Définition
NumSal	Numéro du salarié
NomSal	Nom du salarié
PnomSal	Prénom du salarié
NumPoste	Numéro du poste
DésiPoste	Désignation du poste
<u>IntComp</u>	Intitulé de la compétence
<u>CodeNiv</u>	Code du niveau: A, B ou C



*Notons que le couple (NumSal, IntComp) détermine un et un seul CodeNiv.*

2. Indiquer les incohérences éventuellement admises par la modélisation proposée.

Un salarié peut occuper un poste sans posséder une compétence requise alors qu'il devrait l'avoir avec un certain niveau.

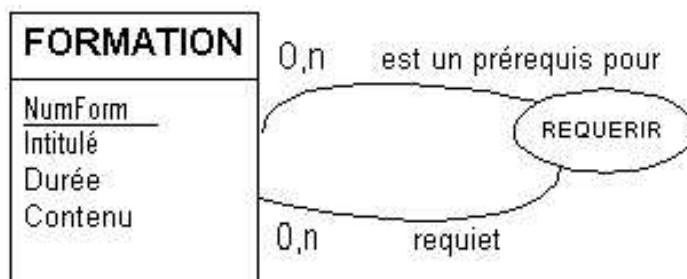


3. Afin de permettre à chaque employé d'évoluer dans son poste ou sur un autre poste, l'entreprise X propose différentes formations. Un extrait du catalogue est présenté ci-dessous:

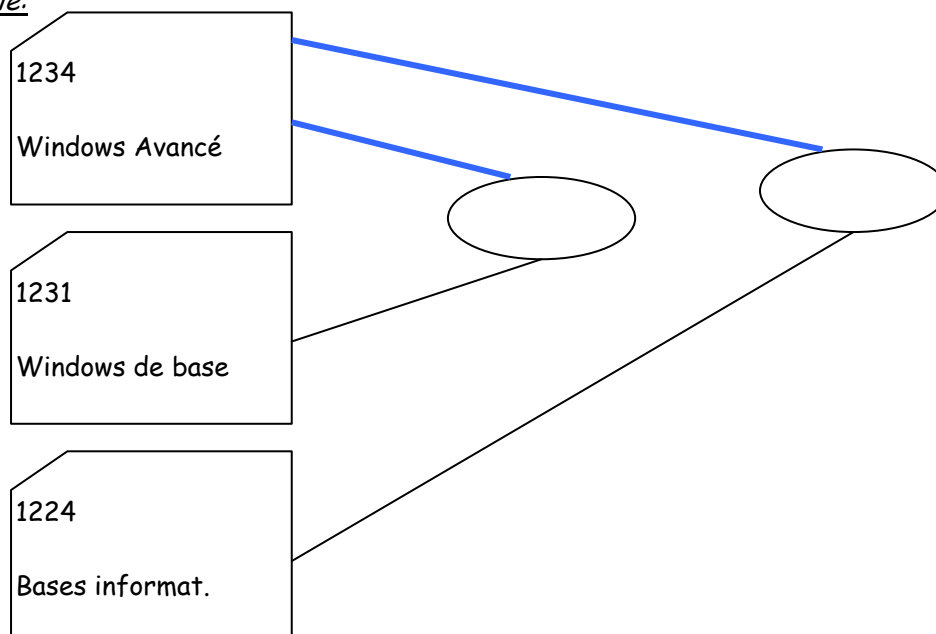
Formation n° 1231
Intitulé : WINDOWS de base
Durée: 5 jours
Contenu : blablabla
Formation(s) prérequis :

Formation n° 1234
Intitulé: WINDOWS Avancé
Durée: 3 jours
Contenu: blablabla
Formations pré-requis: 1231, 1224

Proposer un modèle conceptuel des données permettant d'accueillir l'ensemble des données contenues dans le catalogue de formations.



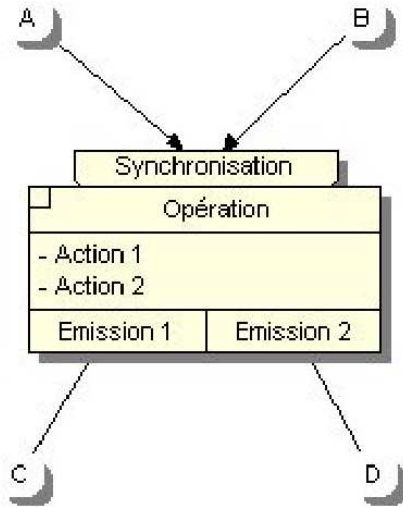
*Exemple:*



## 2.2.6 MCT (Modèle dynamique : prg 2ème année) Modèle Conceptuel des Traitements

Il formalise les événements en fonction des événements extérieurs.

Représentation :

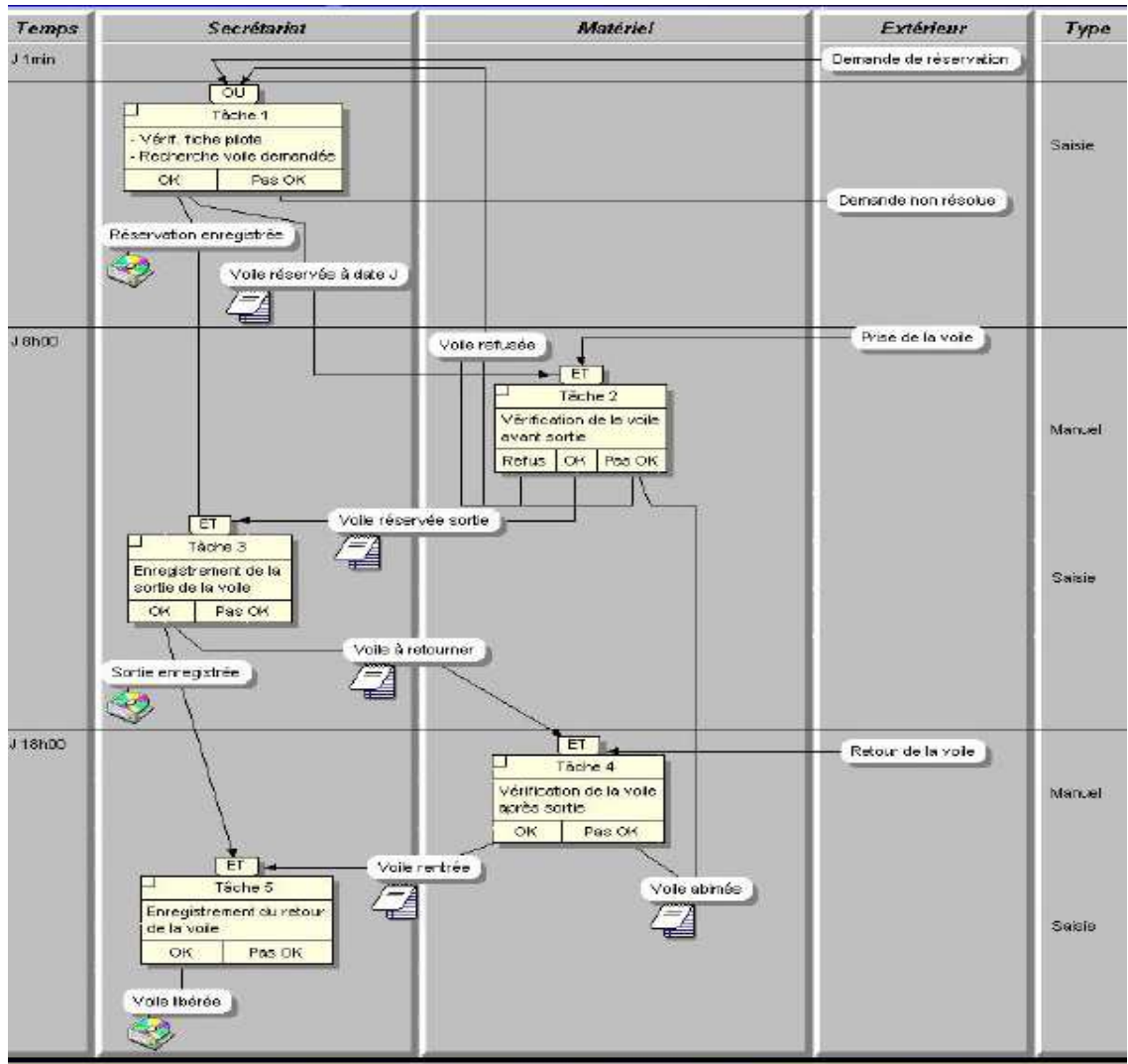


## 2.3 L'analyse (ou formalisation) organisationnelle :

### 2.3.1 MOT Modèle Organisationnel des Traitements (2ème année)

Il est issu du MCT. Il est représenté par un tableau

Exemple :



### **2.3.2 MLD Modèle Logique de Données.**

Il ajoute la notion d'organisation au MCD

Il nécessite la connaissance des moyens disponibles pour manipuler les données

## **2.4 L'analyse (ou formalisation) opérationnelle :**

### **2.4.1 MOPT (2ème année) Modèle organisationnel des traitements.**

Il a pour objectif de préparer au développement :

Définition des normes de développement

Décomposition de chaque application

Définition de cahier des tests

2 démarches : analyse ascendante ou descendante.

### **2.4.2 MPD Modèle Physique des Données.**

Il a pour objectif d'optimiser la gestion des données.